

NAME

zip_fopen_encrypted, **zip_fopen_index_encrypted** - open encrypted file in zip archive for reading

LIBRARY

libzip (-lzip)

SYNOPSIS

```
#include <zip.h>
```

```
zip_file_t *
```

```
zip_fopen_encrypted(zip_t *archive, const char *fname, zip_flags_t flags, const char *password);
```

```
zip_file_t *
```

```
zip_fopen_index_encrypted(zip_t *archive, zip_uint64_t index, zip_flags_t flags,  
    const char *password);
```

DESCRIPTION

The **zip_fopen_encrypted**() function opens the encrypted file name *fname* in *archive* using the password given in the *password* argument. If *password* is NULL or the empty string, the default password is used (see **zip_set_default_password**(3)). The *flags* argument are the same as for **zip_fopen**(3).

The **zip_fopen_index_encrypted**() function opens the file at position *index*, see **zip_fopen_index**(3). These functions are called automatically by **zip_fopen**(3); you only need to call them if you want to specify a non-default password (see **zip_set_default_password**(3)).

RETURN VALUES

Upon successful completion, a *struct zip_file* pointer is returned. Otherwise, NULL is returned and the error code in *archive* is set to indicate the error.

ERRORS

[ZIP_ER_NOPASSWD] No password was provided.

The function **zip_fopen_encrypted**() may also fail and set *zip_err* for any of the errors specified for the routine **zip_fopen**(3).

The function **zip_fopen_index_encrypted**() may also fail and set *zip_err* for any of the errors specified for the routine **zip_fopen_index**(3).

SEE ALSO

libzip(3), **zip_fclose**(3), **zip_fopen**(3), **zip_fread**(3), **zip_get_num_entries**(3), **zip_name_locate**(3)

HISTORY

zip_fopen_encrypted() and **zip_fopen_index_encrypted()** were added in libzip 1.0.

AUTHORS

Dieter Baron <*dillo@nih.at*> and Thomas Klausner <*tk@giga.or.at*>

CAVEATS

The zip file format provides very limited possibility for password verification (a short hash of is compared against one byte in the zip archive). For this reason, reading a file while using an incorrect password may immediately fail with ZIP_ER_WRONGPASSWD, but if the mismatch is not detected, a zlib error may be returned later instead. Since zlib errors can also be caused by broken compressed data, there is no way to make sure if the password was incorrect or if it was correct, but the compressed data was invalid.